Nathaniel Serrano

COS 420

Program 6

3.26.2025

Lab Book

Time spent working with AI: March 25, 8:15 pm - 8:59 pm

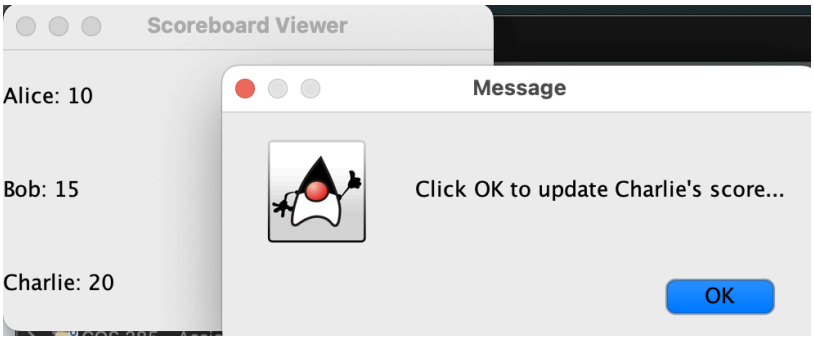
Build a Model

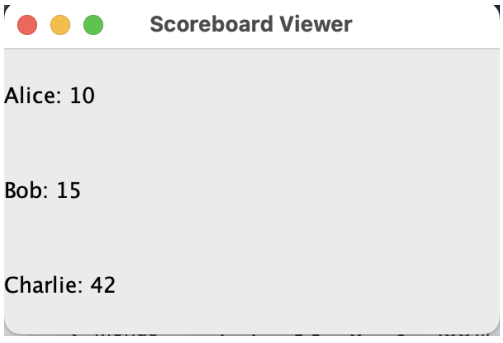
This section of the assignment is a rather simple programming task whose difficulty comes from the time the user spends having to rewrite all methods a PlayerList would use in a Bulldog game. The actual methods themselves are quite simple, with only one line of code each that translates each PlayerList method into ArrayList terms. An AI tool is useful here primarily for the time that it saves. Without the AI tool, I would have to go through my Bulldog game code and look for methods that were used on the player roster, before rewriting them for the PlayerList class one-by-one. An easy task, yes, but it can be tedious, which is not something you want when dealing with deadlines. Chat GPT 4o completed this task less than 10 seconds after I entered the prompt, saving me at least a few minutes. The prompt came from the first two sentences as well as the final sentence in the instructions. After looking through the class to ensure the tool had not missed any important methods, I confirmed that it completed the task with no issues. I did not need to add any edits of my own. Small time-saves like the one that occurred here add up over the course of a project. They represent the true value that AI tools bring: more work completed in

less time. Of course, it is left to the user to ensure the quality of their work remains as good or better than when they worked without the AI tool. It is important to note that this time around, I chose to continue the same session with Chat GPT 4o from the previous assignment, Program 5. This meant that the AI tool already had the full understanding of the codebase and saved me the time of having to share everything with it. In addition, it remembered my request for Javadoc documentation and maintained it throughout this assignment as well.

Build a Viewer

This section of the assignment focused on creating a viewer in the form of a scoreboard that would later be integrated into the Bulldog game. I had Chat GPT 4o accomplish it by prompting it using the given instructions. In addition, I prompted it to provide a main method in which to showcase a demo of the scoreboard changing after interacting with a dialogue menu, which it successfully did without issue. This is shown in the screenshots below.

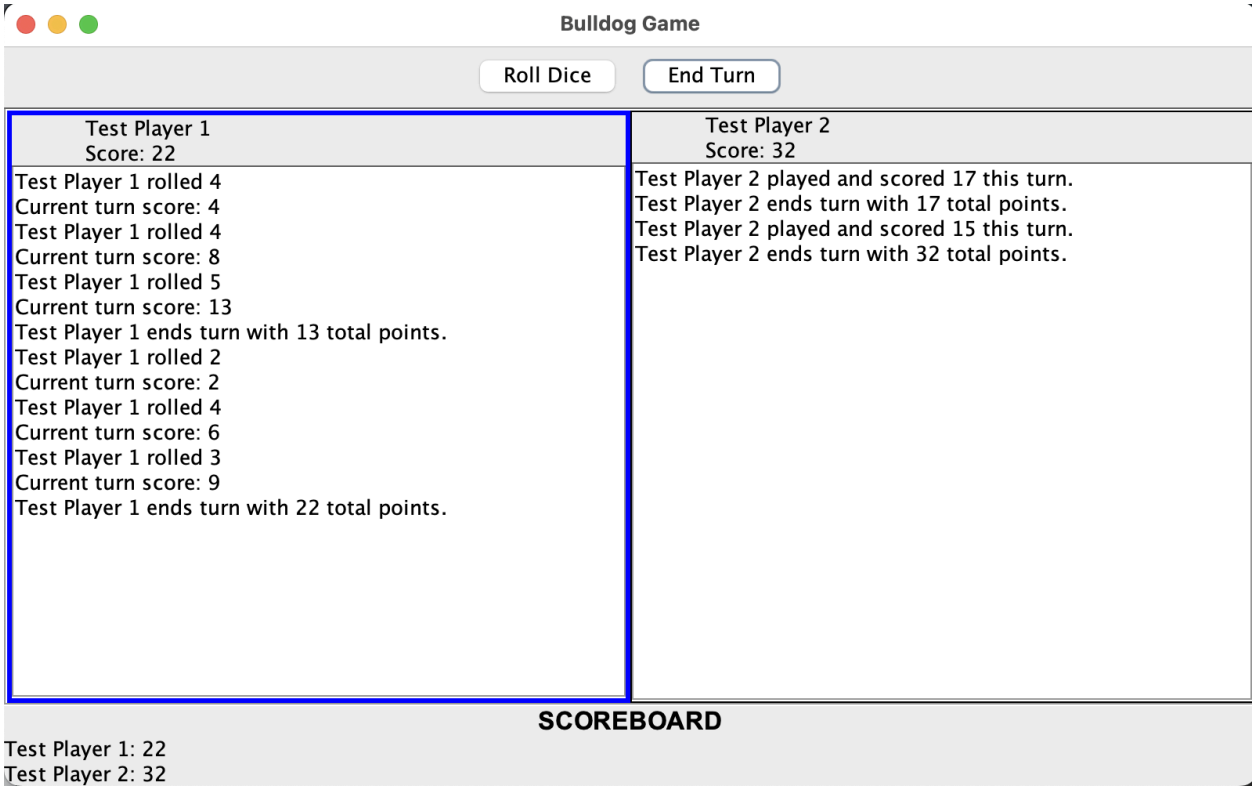




Chat GPT 4o accomplished what was shown in the screenshots by creating a model (PlayerList) and populating it with Players. It created three players and set each of them to have some score value. Then, the dialogue box was displayed, whose message describes what happens when the box is dismissed. Lastly, the Player Charlie's score is changed, which only takes effect after the message box from the viewer has been closed and the viewer has been refreshed. This results in the illusion that the user interacting with the viewer is directly affecting the model, like a controller. The time-save for this section through the use of AI tools was great, since my lacking skills in JSwing would have made for a long and inefficient workflow that would likely be riddled with mistakes. In the end, Chat GPT 4o was able to accomplish this task without requiring any edits to the code from me.

Integration into Bulldog

This final section was focused on integrating the newly created viewer and model into the Bulldog GUI. I chose to do this through two separate prompts due to how big the file concerning Bulldog's GUI is compared to other files in the codebase. If either step resulted in some error, this proactive measure would help determine which addition is responsible for the error. Fortunately, the output from the AI tool did not produce any errors and that proactive measure ended up not being needed. Below is a screenshot showing the scoreboard for the Bulldog game in action. I additionally prompted Chat GPT 4o to add the "SCOREBOARD" label to distinguish the viewer section, which required the label and viewer to be wrapped in a panel that was then added to the GUI. A task that would have been quite difficult for a novice in JSwing like myself was accomplished by Chat GPT 4o upon a simple request.



The time saved from this section was also impressive. Without an AI tool, I would have to comb through the large GUI file myself, and have to contend with learning more JSwing. In addition, figuring out something like the "SCOREBOARD" label would have taken a bit of extra time for me. In the end, like the previous tasks, Chat GPT 4o was able to successfully integrate the model and viewer into the Bulldog GUI without any issues. No additional code or edits were needed from me.

Conclusion

In terms of design patterns and mimicking code, I found that the AI tool was invaluable, simply because it knows more than I do. In order for me to properly code some version of a model or some other design pattern, I have to read up on their definitions, make sure I fully understand what they need and/or do. The AI tool, on the other hand, is trained off of code that

makes up design patterns. In a way, every single piece of code that it generates is a form of mimicry. It has access to the internet in the case it needs to reference something from the web. It is to be expected that such a tool would be good at mimicking code. Above all, the code it generates is generated in a timely manner. I mentioned this previously, but this assignment is indicative of the true value that AI tools bring: more work completed in less time. You as the user are responsible for debugging this work and ensuring it remains high quality. That debugging process combined with the time spent utilizing the AI tool still amounts to a much quicker workflow than one that lacks AI tools, or at least that was the case in this instance. The total time spent working with AI tools for this assignment amounted to only 45 minutes, which I think is an impressive time given the tasks accomplished and the fact that not 100% of that time was spent prompting and coding. I could have spent some additional time on the aesthetics of things like the scoreboard but figured what I had would be sufficient for now. The extra time saved from utilizing AI tools can allow for users to make decisions like that, devoting more time to things like aesthetics or other tasks when they suddenly find themselves with extra time. It is because of its effect on time management that I believe AI tools will be considered a necessity in the coming generations of programming.